Heroku Manual

https://devcenter.heroku.com/articles/how-heroku-works

Bruno ruas

11 de março de 2022

Conteúdo

Capítulo 1

How Heroku Works

1.1 Definindo uma Aplicação

Uma aplicação é a coleção de **código-fonte** escrita em Ruby, Node.js, Java, Python, Clojure, Scala e PHP, alguma **dependency description** que instrui a build system com as informações adicionais sobre as dependências necessárias para a build e run da aplicação e um **Procfile** que contém os comandos nomeados que serão usados na build.

O mecanismo de dependência varia de acordo com a linguagem:

- Ruby → Gemfile
- Python → requirements.textbf
- Node.js → package.json
- Java → pom.xml

1.2 Sabendo o que Executar

Um **requirement** é informar a plataforma quais partes da aplicação são runnable.

O Heroku consegue perceber essas partes dos principais frameworks:

- Ruby on Rails → rails server
- Django → <app>/manage.py runserver
- Node.js → main in package.json

Para algumas aplicações você deve explicitamente declarar o que deve ser executado. Isso é feito em um arquivo texto que vai junto do source code

chamado de **Procfile**.

Cada linha do Procfile declara um **process type** que são os comandos nomeados que devem ser executados no processo de build da aplicação.

Um exemplo de Procfile:

```
web: java -jar lib/foobar.jar $PORT
queue: java -jar lib/queue-processos.jar
```

Esse exemplo declara dois process type. O primeiro chamado `web` que será executado no caminho indicado e o process type `queue`.

Vale a pena, quando for planejar a sua aplicação. Se preocupar com a arquitetura dela. Uma referência que pode ser usada para isso é `https://devcenter.heroku.com/articles/architecting-apps`.